

Compressed Sensing Channel Estimation for OTFS Modulation in Non-Integer Delay-Doppler Domain

Felipe Gómez-Cuba

¹atlanTTic, University of Vigo, Spain. Email: gomezcuba@gts.uvigo.es

Abstract—This paper introduces a Compressed Sensing (CS) estimation scheme for Orthogonal Time Frequency Space (OTFS) channels with sparse multipath. The OTFS waveform represents signals in a two dimensional Delay-Doppler (DD) orthonormal basis. The proposed model does not require the assumption that the delays are integer multiples of the sampling period. The analysis shows that non-integer delay and Doppler shifts in the channel cannot be accurately modelled by integer approximations. An Orthogonal Matching Pursuit with Binary-division Refinement (OMPBR) estimation algorithm is proposed. The proposed estimator finds the best channel approximation over a *continuous* DD dictionary without integer approximations. This results in a significant reduction of the estimation normalized mean squared error with reasonable computational complexity.

Index Terms—OTFS, Compressed Sensing, Channel Estimation

I. INTRODUCTION

Consumer demand for versatile wireless communication systems continues to rise. Release 16 of the Third Generation Partnership Project (3GPP) mobile wireless standard has been completed recently [1], and the 5G mobile standard continues to evolve quickly with new additions in each release. Orthogonal Time Frequency Space (OTFS) is a promising novel waveform proposed recently [2], [3]. The OTFS waveform is based on an explicit delay-Doppler (DD) representation of time-varying multipath channels. Unlike a fading Orthogonal Frequency Division Multiplexing (OFDM) channel, the OTFS channel is quasi-static, i.e., the DD representation of the multipath reflections does not change over long periods of time. Moreover, OTFS is in practice an extension of OFDM with the addition of an internal representation of Doppler (in the sense of a closed group). Thus the new modulation has great backward compatibility with current standards, and the potential to facilitate performance improvements such as Doppler robustness in high speed mobility systems or reducing the perceived channel variability with movement in multipath environments with sparse reflections.

The DD domain is a two-dimensional representation of time-varying signals based on the Discrete Zak Transform (DZT). This mathematical tool has been studied even prior to the proposition of OTFS modulation, for example in polyphase filtering [4]. Hadani *et al* [2] proposed an OTFS implementation based on the cascade of two transforms: first a 2D *Symplectic Fourier Transform* (SFT) converts the DD domain into a time-frequency (TF) grid (similar to a Single-Carrier OFDM (SC-OFDM) scheme in 5G uplink modulations). After the 2D SFT, a second operation termed *Hadamard Transform*

(HT) converts the TF grid into the final time-domain modulated signal. This implementation is not unique, and the DD signal may be converted directly to the time domain using the Inverse DZT (IDZT) as shown in [5], [6]. A derivation of DD modulations from the fundamental principles of sampling is also given in [7]. Several authors have analyzed OTFS transmitter and receiver design [8]–[10], channel estimation [9]–[12], window design [5], [12], etc. In this paper, we highlight three aspects of OTFS channel estimation that have not received sufficient attention in the prior works:

- i) The vast majority of prior literature has assumed the multipath delay can be approximated as an integer multiple of the sampling period [5], [8]–[12]. We model the OTFS DD modulation on a cyclic-prefixed continuous time signal without assuming integer delays in the channel. We show that removing this assumption results in significant differences in the equivalent channel expression.
- ii) Our model employs the direct IDZT synthesis method, rather than the more usual SFT-HT cascade method. We remark that our derivation is independent from the prior result in [6], but both agree in pointing out that non-integer delays modify the channel model significantly.
- iii) Motivated by the observation that non-integer delays play an important role, we propose a Compressed Sensing (CS) channel estimation algorithm for OTFS systems with IDZT architecture that estimates the **continuous** physical DD multipath parameters. Although prior works have applied CS to OTFS channel estimation, the previous references have employed discrete delay dictionaries, resulting in substantially different models.

CS studies the estimation of sparse signals from a limited number of observation samples [13]. By exploiting sparsity, it is possible to develop estimation schemes that outperform conventional techniques. For example, finer delay resolution than expected according to the Shannon-Nyquist sampling constraint is possible in sparse multipath channel estimation. CS has gained interest in recent years as it has enabled to exploit the very large number of antennas and bandwidth in Massive MIMO and mmWave architectures for 5G [1], [14]. The DD representation of signals is sparse in both delay and Doppler dimensions, and the OTFS modulation is a candidate technology for beyond 5G systems. Therefore, the study of CS channel estimation for OTFS is a natural next step for channel estimation in 5G. Moreover, in this paper we show that CS can play a very important role in coping with the differences that

arise in OTFS system models depending on whether or not an assumption is made that delays are integer multiples of the sampling period. For this we develop an extension to OTFS waveforms of the Orthogonal Matching Pursuit with Binary-division Refinement (OMPBR) algorithm first introduced in [14]. Differently from most CS algorithms, OMPBR is not limited in resolution by a discrete dictionary, and can estimate with arbitrary resolution the true continuous values of delay and Doppler multipath parameters. Our results shows that this leads to significant improvement of the estimation error.

The rest of this paper is organized as follows: Section II describes the OTFS channel model. Section III describes the OMPBR estimation algorithm. Section IV validates the results in simulation. And finally, Section V concludes the paper.

A. Notation

Calligraphic letters denote sets. $|\mathcal{A}|$ denotes the cardinality of set \mathcal{A} . Bold uppercase and lowercase letters denote matrices and vectors, respectively. \mathbf{A}^H is the Hermitian and \mathbf{A}^\dagger the Moore-Penrose pseudo-inverse $(\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H$. $\|\mathbf{A}\|_n = \left(\sum_{i,j} |a_{i,j}|^n\right)^{\frac{1}{n}}$ is the ℓ_n norm of \mathbf{A} , and $\|\mathbf{A}\| = \|\mathbf{A}\|_2$.

II. SYSTEM MODEL

We begin by considering a discrete-time signal $x[n]$ of length L transmitted in continuous time using a pulse $p(t)$

$$x(t) = \sum_{n=0}^{L-1} x[n]p(t - nT_s), \quad (1)$$

over a continuous time-variant channel with output satisfying

$$y(t) = \mathcal{H}(x(t)) = \sum_{i \in \mathcal{P}} a_i e^{-j2\pi\nu_i t} x(t - \tau_i) + r(t) \quad (2)$$

where $r(t) \sim \mathcal{CN}(0, \sigma_z^2)$ is AWGN and \mathcal{P} is a set of paths characterized by their complex gain $a_i \in \mathbb{C}$, Doppler shift $\nu_i \in [-\frac{V\Delta f}{2}, \frac{V\Delta f}{2}]$ and delay $\tau_i \in [0, DT_s)$. We assume D and V are integers selected to capture the maximum delay and Doppler of the channel, normalized with regard to the symbol period T_s and inverse message duration $\Delta f = \frac{1}{LT_s}$. Without loss of generality, we assume the symbol length satisfies $L = DV$. In typical systems it is common to adopt easily divisible values of L such as powers of 2 [1]. Thus, the case $L > DV$ can be easily included in our model adopting suitable integers $D' > D$ and $V' > V$ satisfying $L = V'D'$.

In order to convert (2) into a circular convolution we consider the *periodic extension* of $x[n]$, denoted $x_p[n] = x[n \bmod L]$. This results in the continuous-time transmitted signal

$$\begin{aligned} x_p(t) &= \sum_{n=0}^{L-1} x[n] \sum_{m=-\infty}^{\infty} p(t - nT_s - mLT_s) \\ &= \sum_{n=0}^{L-1} x[n] g_L(t/T_s - n). \end{aligned} \quad (3)$$

Since sinc interpolation and sampling can be regarded as dual operations when T_s is adequately chosen, without loss of generality in this paper we assume $p(t) = \text{sinc}(t/T_s)$, and

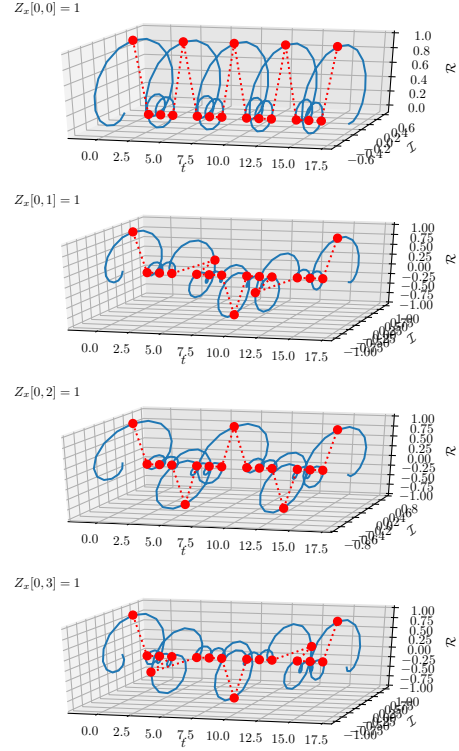


Fig. 1. OTFS modulation orthonormal basis representation

thus $g_L(t) = e^{-j(L-1)t} \frac{\sin(2\pi Lt)}{\sin(2\pi t)}$ is a *Dirichlet* periodical pulse train [4], [7], [12]. Other pulses and windows can be modeled in discrete time, applied on $x_p[n]$, as long as we interpret T_s as the *sampling* period, not a *modulation symbol* period.

We define the DZT of $x[n]$ [4] as

$$Z_x[d, v] = \sum_{u=0}^{V-1} x[d + uD] e^{-j2\pi \frac{uv}{V}}.$$

noting that the DZT is superjective and $Z_x[d, v] = Z_{x_p}[d, v]$. In other words, the *Inverse Discrete Zak Transform* (IDZT) [4] outputs the periodic extension of the discrete signal

$$x_p[n] = \sum_{v=0}^{V-1} Z_x[n \bmod D, v] e^{j2\pi \frac{1n/D1v}{V}}. \quad (4)$$

However, since $x[n]$ is limited to the interval $\{0 \dots L-1\}$ we can always unequivocally reconstruct $x[n]$ from $Z_x[d, v]$.

To perform the OTFS modulation, information is encoded in the DD domain coefficients $Z_x[d, v]$ [5]. The cyclic signal $x_p(t)$ is a linear combination of the OTFS coefficients, multiplied by an orthonormal basis formed by the Doppler shifted dirichlet pulses $e^{-j2\pi \frac{v}{V} t} g_D(t/T_s - d)$. Fig. 1 illustrates the orthonormal basis used in OTFS modulation (4), for $D = V = 4$ and $d = 0$, with all Doppler shifts $v \in \{0 \dots V-1\}$.

In practical implementations we would desire to synthesize a finite transmission. In order to replicate the circular convolution for a limited time, the transmitter sends the signal

$$x_{CP}(t) = \begin{cases} x_p(t) & t \in [-DT_s, LT_s] \\ 0 & \text{otherwise,} \end{cases} \quad (5)$$

noting that $x_{CP}(t)$ is related to $x[n]$ by (3) and not (1). Reciprocally, the receiver applies a rectangular window for the interval $t \in [0, LT_s]$ to (2), producing

$$y_{CP}(t) = \begin{cases} \mathcal{H}(x_p(t)) & \forall t \in [0, LT_s] \\ 0 & \text{otherwise.} \end{cases} \quad (6)$$

Finally, thanks to the cyclic prefix in $x_{CP}(t)$, when the receiver samples $y_{CP}(t)$ with period T_s and computes its DZT, the result is an Equivalent Zak Channel (EZC) with output

$$Z_y[d, v] = \sum_{u'=0}^{V-1} \sum_{d'=0}^{D-1} h[d, v, d', v'] Z_x[d', v'] + Z_r[d, v] \quad (7)$$

where $Z_r[d, v]$ is the Zak transform of the noise, which is still AWGN, and the channel gains are given by

$$h[d, v, d', v'] = \sum_{i \in \mathcal{P}} a_i e^{-j\nu_i d/D} g_D(d - d' - \tau_i/T_s) \times g_V(v - v' - \nu_i/\Delta f) e^{-j2\pi \frac{v}{V} \lfloor \frac{d-d'}{D} \rfloor}. \quad (8)$$

Here we remark that the EZC output (7) cannot, in general, be expressed as a 2D convolution of $Z_h[d, v]$ and $Z_x[d, v]$, where $Z_h[d, v]$ is a DZT representation of the channel. There are two exceptions: when either all normalized Doppler coefficients $\nu_i/\Delta f$, or all normalized delay coefficients τ_i/T_s , are integers. In either of these two special integer cases, we get

$$h[d, v, d', v'] = Z_h[d - d', v - v'] \quad (9)$$

where $Z_h[d, v]$ is the DZT of a Dirichlet channel probe

$$h[n] = \mathcal{H}(g_L(t/T_s))|_{t=nT_s} = \sum_{i \in \mathcal{P}} a_i e^{-j2\pi \nu_i n T_s} g_L(n - \tau_i/T_s).$$

We can define the vector $\mathbf{x} \in \mathbb{C}^L$ such that its ℓ -th term contains $Z_x[\ell \bmod D, \lfloor \ell/D \rfloor]$, and likewise $\mathbf{y}, \mathbf{z} \in \mathbb{C}^L$. Moreover, defining the matrix $\mathbf{H} \in \mathbb{C}^{L \times L}$ with $h[\ell \bmod D, \lfloor \ell/D \rfloor, \ell' \bmod D, \lfloor \ell'/D \rfloor]$ in its ℓ -th row and ℓ' -th column, the EZC (7) can be fully written as

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{z}. \quad (10)$$

Comparison of (7) and (10) with a classic CP-OFDM system for time-invariant channels reveals some similarities and differences. In a CP-OFDM system with a L -DFT and an infinite cyclic extension of the transmitted signal, the Dirichlet pulse would also be observed in the discrete channel impulse response of length D with non-integer multipath delays. Moreover, a second Dirichlet pulse would also arise as an interpolator in frequency, since the L subcarrier gains would be correlated when $D < L$.

If we set $V = L$ and $D = 1$ the OTFS scheme fully becomes CP-OFDM. More generally, for $D > 1$ the IDZT/DZT transforms correspond to the time-domain half of the decimation-in-time FFT butterfly algorithm. This makes the implementation of the DZT computation very efficient.

In CP-OFDM channels, the cyclic extension enables writing the channel as a scalar multiplication in each subcarrier.

However, due to non-integer Doppler, in the Zak channel we must take into account a grand total of L^2 multiplications to compute the first term of the EZC (7). Owing to the highly structured expression (8), a naïve and wasteful direct estimation of all L^2 elements of the matrix \mathbf{H} would not be necessary. Instead, we assume that the number of multipath components is $P = |\mathcal{P}|$ satisfying $P \ll L$. Therefore, we can estimate the explicit multipath characteristics of the channel using CS methods as detailed in the next section. In a practical system, we would consider the finite implementation (5) repeatedly transmitting consecutive blocks, each carrying L symbols \mathbf{x} on a physical signal of duration $(L + D)T_s$. Despite the time-domain channel being time-variant, the explicit multipath parameters are quasi-static. Thus, it suffices that the transmitter sends a known pilot \mathbf{x} in the first block of each session.

III. CS CHANNEL ESTIMATION ALGORITHM

To design the CS estimator [13] we begin by defining the set of matrices $\Upsilon(\tau, \nu)$ for any value of τ and ν , such that

$$\{\Upsilon(\tau, \nu)\}_{d+vD, d'+v'D} = e^{-j\nu_i d/D} g_D(d - d' - \tau/T_s) \times g_V(v - v' - \nu/\Delta f) e^{-j2\pi \frac{v}{V} \lfloor \frac{d-d'}{D} \rfloor}. \quad (11)$$

We multiply this matrix by the known pilot \mathbf{x} to obtain a generalized basis vector $\phi(\tau, \nu) = \Upsilon(\tau, \nu)\mathbf{x}$. We also define the 2D continuous channel spreading function as follows:

$$A_{\mathcal{P}}(\tau, \nu) = \sum_{i \in \mathcal{P}} a_i \delta(\tau - \tau_i, \nu - \nu_i).$$

So we may substitute (11) and (8) into (10) [6], [12], producing

$$\begin{aligned} \mathbf{y} &= \sum_{i \in \mathcal{P}} a_i \Upsilon(\tau_i, \nu_i) \mathbf{x} + \mathbf{z} \\ &= \sum_{i \in \mathcal{P}} a_i \phi(\tau_i, \nu_i) + \mathbf{z} \\ &= \int_0^{DT_s} \int_0^{V\Delta f} \phi(\tau, \nu) A_{\mathcal{P}}(\tau, \nu) d\nu d\tau + \mathbf{z}. \end{aligned} \quad (12)$$

Where the interval $[\frac{V\Delta f}{2}, V\Delta f)$ represents the negative Doppler ranges $[-\frac{V\Delta f}{2}, 0)$, due to the periodicity of $g_V(\cdot)$.

Since the integral in (12) may be challenging, it is frequent to design CS schemes using a discrete **dictionary** of values [13]. In our case, we define dictionaries of K_τ and K_ν evenly spaced points in the intervals $[0, DT_s)$ and $[0, V\Delta f)$, respectively. The dictionary is thus the cartesian product $\mathcal{D}_\tau \times \mathcal{D}_\nu$, where

$$\mathcal{D}_\tau = \{0 \dots \frac{K_\tau - 1}{K_\tau} DT_s\}, \quad (13)$$

and

$$\mathcal{D}_\nu = \{0 \dots \frac{K_\nu - 1}{K_\nu} V\Delta f\}. \quad (14)$$

Finally, we define the dictionary basis matrix containing all possible basis vectors in the dictionary as

$$\Phi_{\mathcal{D}_\tau \times \mathcal{D}_\nu} = (\phi(0, 0), \dots, \phi(\frac{K_\tau - 1}{K_\tau} DT_s, \frac{K_\nu - 1}{K_\nu} V\Delta f)),$$

and an approximation of the channel output can be written as

$$\mathbf{y} \simeq \Phi_{\mathcal{D}_\tau \times \mathcal{D}_\nu} \mathbf{a} + \mathbf{z} \quad (15)$$

where \mathbf{a} is a sparse vector with non-zero values in the indices that correspond to existing paths in the channel.

We remark that the dictionary dimensions K_τ and K_ν may be greater than D and V , respectively. In CS, this is referred to as an *overcomplete* dictionary. Overcomplete dictionaries enable *superresolution*, i.e., to resolve the support parameters, in our case delay and Doppler, with more resolution than it would be possible in non-sparse estimation according to the Shannon-Nyquist sampling constraints [13]. In fact, we have noted that the channel expression (7) cannot be expressed in the form (9) when delay and Doppler are both fractional. Thus the use of orthogonal dictionaries without superresolution, with $K_\tau = D$ and $K_\nu = V$, will lead to significant errors.

When we assume $K_\tau > D$ and $K_\nu > V$, the matrix $\Phi_{\mathcal{D}_\tau \times \mathcal{D}_\nu}$ in (15) is wide and does not have a pseudoinverse. Thus we cannot resort to well known methods such as Least Squares (LS) to estimate \mathbf{a} . Therefore, the sparsity of \mathbf{a} must be exploited. The fundamental form of a CS estimator of \mathbf{a} is

$$\hat{\mathbf{a}} = \arg \min \|\mathbf{a}\|_0 \text{ s.t. } \|\mathbf{y} - \Phi_{\mathcal{D}_\tau \times \mathcal{D}_\nu} \mathbf{a}\|_2 \leq \xi \quad (16)$$

where ξ is a tuning parameter to account for the noise. As problem (16) is combinatorial, there are two general families of CS algorithms to convert (16) into a tractable problem: greedy approximations of the combinatorial, or problem relaxations by substituting the ℓ_0 norm with an ℓ_1 norm [13]. A second distinction can be made between algorithms that assume the number of non-zero elements of \mathbf{a} , P , is known and those that do not. Among greedy algorithms with unknown P , Orthogonal Matching Pursuit (OMP) variants are well known. Greedy algorithms with known P include Compressive Sampling Matching Pursuit (CoSaMP). ℓ_1 algorithms with unknown P include Basis Pursuit De Noising (BPDN) and the Dantzig-Selector. Finally, the Least Absolute Shrinkage and Selection Operator (LASSO) solves a dual problem with known P , minimizing the ℓ_2 norm with an ℓ_1 constraint.

Since the number of multipath reflections can change, in this paper we assume P is not known. Moreover, to develop our extension using the continuous sparse model (12), we must adopt a greedy algorithm. Therefore, of all algorithms in literature, the most relevant baseline for our study is OMP

Algorithm 1 Orthogonal Matching Pursuit

- 1: Initialization $\hat{\mathcal{P}} = \emptyset$, $\mathbf{r} = \mathbf{y}$
 - 2: **while** $\|\mathbf{r}\|^2 > \xi$ **do**
 - 3: $(\hat{\tau}, \hat{\nu}) = \arg \max_{\tau \in \mathcal{D}_\tau, \nu \in \mathcal{D}_\nu} \|\phi(\tau, \nu)^H \mathbf{r}\|$
 - 4: $\hat{\mathcal{P}} = (\hat{\tau}, \hat{\nu}) \cup \hat{\mathcal{P}}$
 - 5: $\hat{\mathbf{a}} = \Phi_{\hat{\mathcal{P}}}^\dagger \mathbf{y}$
 - 6: $\mathbf{r} = \mathbf{y} - \Phi_{\hat{\mathcal{P}}} \hat{\mathbf{a}}$
 - 7: **end while**
 - 8: **Output** $\hat{\mathcal{P}}$, $\hat{\mathbf{a}}$
-

(Alg. 1). Here, we use the notation $\hat{\mathcal{P}}$ to represent the estimated set of delay and Doppler support values, which is increased by exactly one element in each iteration. Moreover, we use the matrix notation $\Phi_{\hat{\mathcal{P}}}$ to represent the support matrix associated with $\hat{\mathcal{P}}$, that is, each column of $\Phi_{\hat{\mathcal{P}}}$ is the vector $\phi(\hat{\tau}_j, \hat{\nu}_j)$ associated with the j -th element of $\hat{\mathcal{P}}$. As the greedy algorithm increases the support, it builds $\Phi_{\hat{\mathcal{P}}}$ as a tall matrix containing a subset of the columns of Φ , such that the LS operation in line 5 is possible. In each iteration, the residual \mathbf{r} contains the part of \mathbf{y} that is orthogonal to the current basis $\Phi_{\hat{\mathcal{P}}}$ (line 6). Finally, the greedy algorithm adds an element to $\hat{\mathcal{P}}$ in each iteration (line 4), using a maximum residual correlation criterion (line 3), until the distance constraint ξ is satisfied (line 2). The number of iterations equals the estimated number of paths $|\hat{\mathcal{P}}|$.

To choose the tuning parameter ξ we study the Normalized Mean Squared Error (NMSE) expressed as follows:

$$NMSE = \mathbb{E}_{\mathbf{H}} \left[\frac{\|\mathbf{H} - \sum_{j \in \hat{\mathcal{P}}} \hat{a}_j \Upsilon(\hat{\tau}_j, \hat{\nu}_j)\|^2}{\|\mathbf{H}\|^2} \right] \quad (17)$$

$$\simeq \mathbb{E}_{\mathbf{H}} \left[\frac{\|\mathbf{y} - \Phi_{\hat{\mathcal{P}}} \hat{\mathbf{a}}\|^2}{\|\mathbf{H}\mathbf{x}\|^2} \right] \quad (18)$$

$$= \mathbb{E}_{\mathbf{H}} \left[\frac{\|(\mathbf{I} - \Phi_{\hat{\mathcal{P}}} \Phi_{\hat{\mathcal{P}}}^\dagger)(\Phi_{\hat{\mathcal{P}}} \mathbf{a} + \mathbf{z})\|^2}{\|\mathbf{H}\mathbf{x}\|^2} \right] \quad (19)$$

$$\geq \mathbb{E}_{\mathbf{H}} \left[\frac{\|\Xi_{\hat{\mathcal{P}}} \Phi_{\hat{\mathcal{P}}} \mathbf{a}\|^2}{\|\mathbf{H}\mathbf{x}\|^2} \right] + |\hat{\mathcal{P}}| \sigma^2 \mathbb{E}_{\mathbf{H}} [\|\mathbf{H}\mathbf{x}\|^{-2}] \quad (20)$$

where (18) assumes that the average energy of the matrix \mathbf{H} is symmetrically distributed in all directions, and we define $\Xi_{\hat{\mathcal{P}}} = \mathbf{I} - \Phi_{\hat{\mathcal{P}}} \Phi_{\hat{\mathcal{P}}}^\dagger$. From the lower bound (20) we note that the error contains two terms. The first term represents the projection of the true channel orthogonal to the basis $\Phi_{\hat{\mathcal{P}}}$, and decreases in each iteration. The second term represents the projection of the noise over $\Phi_{\hat{\mathcal{P}}}$, and grows in each iteration. Therefore, a natural choice for ξ is making the algorithm stop when “continuing more iterations would capture more noise than channel”, which results in $\|\mathbf{r}\|^2 < L\sigma^2 \triangleq \xi$ [14].

The classic OMP algorithm displays three shortcomings:

- i) It is based on a discrete approximation (15) of a continuous truth (12). We have argued that using (8) in (7) is not the same as using (9). This means that, in OTFS, quantized multipath integers can severely misrepresent the channel.
- ii) Even though the dictionary sizes K_τ and K_ν can be increased to mitigate problem *i*), this results in a linear growth of the complexity of Line 3, which is equivalent to $\Phi_{\mathcal{D}_\tau \times \mathcal{D}_\nu}^H \mathbf{r}$, that is, $K_\tau K_\nu L$ products.
- iii) Whenever \mathbf{x} changes the matrix $\Phi_{\mathcal{D}_\tau \times \mathcal{D}_\nu}$ needs to be precomputed. Performing $K_\tau K_\nu$ times the operation $\phi(\tau, \nu) = \Upsilon(\tau, \nu) \mathbf{x}$ results in $K_\tau K_\nu L^2$ products.

Due to the above, we introduce the OMPBR algorithm (Alg. 2). The main goal of OMPBR is to reduce the dictionary size in line 3 of Alg. 1. For this we observe that the correlation $f(\tau, \nu) = \|\phi(\tau, \nu)^H \mathbf{r}\|$ is not concave in the full search space $[0, DT_s) \times [0, V\Delta f)$, but it can be assumed to be locally symmetric in small regions near the maximum. Therefore, we

perform a two step search of the correlation maximum: first, we search for the best “local bin” of size $T_s \times \Delta f$ in line 3. This search uses an auxiliary dictionary with $D \times V$ integer elements, which can be accelerated using (9). Second, we identify the best small delay and Doppler offsets contained in the “local bin”, in line 4. This search is implemented using the Binary-division Refinement Alg. 3. When $f(\tau, \nu)$ is symmetric around a local maximum, BR guarantees a distance $2^{-N_{ref}}$ to the true maximum in N_{ref} iterations [14].

Algorithm 2 OMP with Binary division Refinement (OMPBR)

- 1: Initialization $\hat{\mathcal{P}} = \emptyset$, $\mathbf{r} = \mathbf{y}$
 - 2: **while** $\|\mathbf{r}\| > \xi$ **do**
 - 3: $(\hat{d}, \hat{\nu}) = \arg \max_{\substack{d \in \{0 \dots D-1\} \\ v \in \{0 \dots V-1\}}} \|\phi(dT_s, v\Delta f)^H \mathbf{r}\|$
 - 4: $(\hat{\mu}_\tau, \hat{\mu}_\nu) = \arg \max_{\mu_\tau, \mu_\nu \in [-\frac{1}{2}, \frac{1}{2}]} \|\phi((\hat{d} + \mu_\tau)T_s, (\hat{\nu} + \mu_\nu)\Delta f)^H \mathbf{r}\|$
 - 5: $\hat{\mathcal{P}} = ((\hat{d} + \hat{\mu}_\tau)T_s, (\hat{\nu} + \hat{\mu}_\nu)\Delta f) \cup \hat{\mathcal{P}}$
 - 6: $\hat{\mathbf{a}} = \Phi_{\hat{\mathcal{P}}}^\dagger \mathbf{y}$
 - 7: $\mathbf{r} = \mathbf{y} - \Phi_{\hat{\mathcal{P}}} \hat{\mathbf{a}}$
 - 8: **end while**
 - 9: **Output** $\hat{\mathcal{T}}, \hat{\mathcal{V}}, \hat{\mathbf{a}}$
-

BR can also be employed to refine OMP decisions for an overcomplete auxiliary dictionary, although this drops the possibility of accelerating line 3 for integer dictionaries. OMPBR can be regarded as an approximation heuristic to classic OMP with an extremely large dictionary. By observing (13) and (14), if we define the superresolution factors $\kappa_\tau = \frac{2^{N_{ref}} K_\tau}{D}$ and $\kappa_\nu = \frac{2^{N_{ref}} K_\nu}{V}$, it is clear that the complexity of OMPBR scales much better with resolution. OMP ($N_{ref} = 0$) finds the global correlation maximums with linear scaling in resolution $\Theta(\kappa_\tau \kappa_\nu)$, whereas OMPBR provides a good approximation of correlation decisions with only logarithmic scaling $\Theta(\log(\kappa_\tau \kappa_\nu))$.

Algorithm 3 Binary division Refinement (BR)

- 1: **Input:** 2D function $f(x, y)$, initial point (x_o, y_o)
 - 2: Initialize $x_{up} = x_o + 0.5$, $x_{down} = x_o - 0.5$,
 - 3: $y_{up} = y_o + 0.5$, and $y_{down} = y_o - 0.5$
 - 4: **for** N_{ref} iterations **do**
 - 5: $x_{mid} = (x_{up} + x_{down})/2$, $y_{mid} = (y_{up} + y_{down})/2$
 - 6: **switch** (Select maximum and halve search region)
 - 7: **case** $f(x_{up}, y_{up})$ is maximum:
 - 8: $x_{down} = x_{mid}$, $y_{down} = y_{mid}$
 - 9: **case** $f(x_{down}, y_{up})$ is maximum:
 - 10: $x_{up} = x_{mid}$, $y_{down} = y_{mid}$
 - 11: **case** $f(x_{up}, y_{down})$ is maximum:
 - 12: $x_{down} = x_{mid}$, $y_{up} = y_{mid}$
 - 13: **case** $f(x_{down}, y_{down})$ is maximum:
 - 14: $x_{up} = x_{mid}$, $y_{up} = y_{mid}$
 - 15: **end switch**
 - 16: **end for**
 - 17: **Output** (x_{mid}, y_{mid})
-

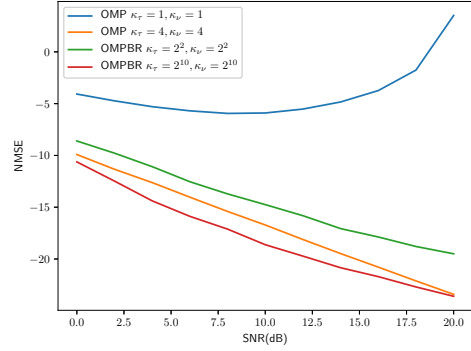


Fig. 2. Channel NMSE

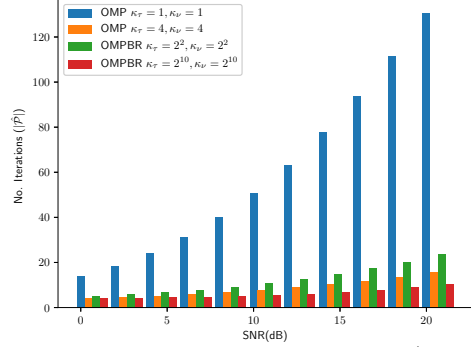


Fig. 3. Number of estimated paths $|\hat{\mathcal{P}}|$

IV. SIMULATIONS

We simulate an OTFS system with dimensions $D=V=16$, modelled as in (7)-(8). We generate 1000 random multipath channels with $P = 3$ independent reflections with normalized distributions $\frac{\tau_i}{T_s} \sim U(0, D)$, $\frac{\nu_i}{\Delta f} \sim U(0, V)$, $\bar{a}_i \sim \mathcal{CN}(0, 1)$, and finally normalize the gains as $a_i = \frac{\bar{a}_i}{\sqrt{\sum_{i \in \mathcal{P}} |\bar{a}_i|^2}}$. The pilot signals are random i.i.d. Gaussian sequences $\mathbf{x} \sim \mathcal{CN}(0, \mathbf{I}_L)$.

Fig. 2 represents the average NMSE evaluated as (17) without approximations, and Fig. 3 represents the number of estimated coefficients. One key observation is that OMP with $K_\tau = D$ and $K_\nu = V$ performs very poorly. In fact, its NMSE does not even decrease with SNR. The explanation is that, as we noted, expressions (8) and (9) are theoretically different. The greedy algorithm runs until it reaches the maximum number of iterations without ever converging and, since each iteration captures some noise, this results in large NMSE. Next, we consider OMP with $K_\tau = 4D$ and $K_\nu = 4V$. The NMSE of this estimator is significantly improved, and the number of estimated paths is reasonable. A slight over-estimation of the number of paths is acceptable as discrete dictionaries can only approximate (8) so much. However, the use of a very large dictionary size can be computationally demanding. Next, we consider the case of OMPBR with $K_\tau = D$ and $K_\nu = V$ and $N_{ref} = 2$. This estimator has the same search space and resolution as OMP with $\kappa_\tau = \kappa_\nu = 4$, but stores a much smaller dictionary in memory. Remarkably, even though an orthogonal auxiliary dictionary is used, OMPBR performs quite well and does not fail catastrophically as OMP with the orthogonal dictionary. Since the BR search is local its

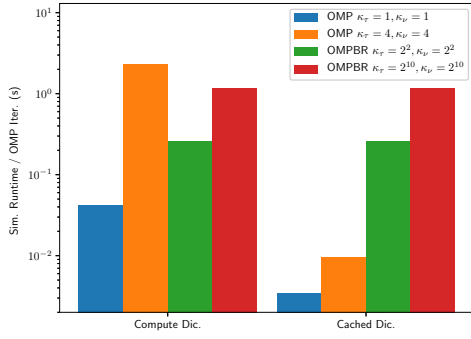


Fig. 4. Total simulation runtime

NMSE is a bit worse than OMP with the same superresolution factors $\kappa_\tau = \kappa_\nu = 4$. The number of estimated paths is slightly higher than the overcomplete OMP case as well. Finally, OMPBR with $K_\tau = D$ and $K_\nu = V$ and $N_{ref} = 10$ has a superresolution factor of $\kappa_\tau = \kappa_\nu = 1024$, which would be impractical with OMP, as it would require 10^6 times more run time and memory. With an increase in complexity that only scales logarithmically, the 2D search space is increased by six orders of magnitude, and the NMSE is significantly improved compared with either OMPBR or OMP with coarser resolution. The number of necessary iterations ($|\hat{\mathcal{P}}|$) is also significantly lower than OMP with $\kappa_\tau = \kappa_\nu = 4$.

To discuss the computational complexity in detail, we depict the mean simulation runtime per iteration in Fig. 4 (using a Dell XPS 13 2019 laptop). Our simulator implements a cache system to avoid precomputing the same matrix $\Phi_{D_\tau \times D_\nu}$ twice. We first focus on the points when the cache was not employed (8% of the total). The time to generate the matrix $\Phi_{D_\tau \times D_\nu}$ and load it in memory is counted in the total run time of Algs. 1 and 2 and divided by $|\hat{\mathcal{P}}|$. In this case, the run time of OMP with $\kappa_\tau = \kappa_\nu = 1$ versus $\kappa_\tau = \kappa_\nu = 4$ grew by a factor of 16, as we have predicted. Moreover, the run-time of OMP with $\kappa_\tau = \kappa_\nu = 4$ was an order of magnitude greater than OMPBR for the same resolution. Finally, the run time of OMPBR with $\kappa_\tau = \kappa_\nu = 1024$ was 2.5 times higher than OMPBR with $\kappa_\tau = \kappa_\nu = 4$, confirming our analysis. Remarkably, OMPBR with $\kappa_\tau = \kappa_\nu = 1024$ is faster than OMP with $\kappa_\tau = \kappa_\nu = 4$, in addition to achieving better channel estimations.

When the matrix $\Phi_{D_\tau \times D_\nu}$ is cached, the run-time comparison changes significantly. The run time of classic OMP decreases by two orders of magnitude, whereas BR does not benefit from caching. Therefore, for moderate values of κ_τ and κ_ν , for the subset of systems with unchanging pilot patterns and a large memory, classic OMP is still preferable over OMPBR in terms of complexity. Still, OMP with a super-resolution factor of $\kappa_\tau = \kappa_\nu = 1024$ would be impossible to implement. OMPBR displays the better scaling and is the only estimator that supports extremely large values of κ_τ and κ_ν , approximating the continuous truth of the channel (12).

V. CONCLUSIONS AND FUTURE WORK

The OTFS waveform allows to represent signals in a sparse Delay-Doppler domain representation. In sparse multipath sce-

narios, the channel output may be expressed as a sum of paths with individual delay and Doppler parameters. Our analysis shows that the channel response displays intrinsic differences when the delay is not an integer multiple of the sampling period and the Doppler shift is not an integer multiple of the frequency parameter $\Delta f = \frac{1}{LT_s}$. Compressed Sensing can enable significant improvements in sparse channel estimation. However, prior work on OTFS CS channel estimation has assumed that the delays are approximately integer multiples of the sampling period. In this paper, we developed a CS channel estimation scheme for OTFS systems that can model the *continuous* delay and Doppler domain. Moreover, we have shown that in OTFS there are intrinsic differences when the delay coefficients are not integer, and the use of integer approximations results in catastrophic channel estimation errors. As a result, we propose an OMPBR algorithm that estimates delay and Doppler on a continuous domain. This significantly reduces the error in OTFS channel estimation with reasonable complexity. In future work we plan to extend our results to multiple-antenna joint sparse angular, delay and Doppler estimation problems and multi-user locations.

REFERENCES

- [1] 3GPP, “3rd Generation Partnership Project; Technical Specification Group Radio Access Network; NR; NR and NG-RAN Overall Description; Stage 2 (Release 16),” *ETSI TS 38.300*, no. 16.5.0, 2021.
- [2] R. Hadani, S. Rakib, M. Tsatsanis, A. Monk, A. J. Goldsmith, A. F. Molisch, and R. Calderbank, “Orthogonal time frequency space modulation,” in *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, 2017, pp. 1–6.
- [3] Z. Wei, W. Yuan, S. Li, J. Yuan, G. Bharatula, R. Hadani, and L. Hanzo, “Orthogonal Time-Frequency Space Modulation: A Promising Next-Generation Waveform,” *IEEE Wireless Communications*, pp. 1–9, 2021.
- [4] H. Bolcskei and F. Hlawatsch, “Discrete Zak transforms, polyphase transforms, and applications,” *IEEE Transactions on Signal Processing*, vol. 45, no. 4, pp. 851–866, 1997.
- [5] P. Raviteja, Y. Hong, E. Viterbo, and E. Biglieri, “Practical pulse-shaping waveforms for reduced-cyclic-prefix OTFS,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 1, pp. 957–961, 2019.
- [6] S. K. Mohammed, “Time-domain to delay-Doppler domain conversion of OTFS signals in very high mobility scenarios,” *IEEE Transactions on Vehicular Technology*, 2021.
- [7] —, “Derivation of OTFS modulation from first principles,” *IEEE Transactions on Vehicular Technology*, 2021.
- [8] V. S Bhat, S. G. Dayanand, and A. Chockalingam, “Performance analysis of OTFS modulation with receive antenna selection,” *IEEE Transactions on Vehicular Technology*, vol. 70, no. 4, pp. 3382–3395, 2021.
- [9] M. Kollengode Ramachandran and A. Chockalingam, “MIMO-OTFS in high-Doppler fading channels: signal detection and channel estimation,” in *IEEE Global Communications Conference (GLOBECOM)*, 2018.
- [10] M. Li, S. Zhang, F. Gao, P. Fan, and O. A. Dobre, “A new path division multiple access for the Massive MIMO-OTFS networks,” *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 4, pp. 903–918, 2021.
- [11] P. Raviteja, K. T. Phan, and Y. Hong, “Embedded pilot-aided channel estimation for OTFS in delay–Doppler channels,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 5, pp. 4906–4917, 2019.
- [12] Z. Wei, W. Yuan, S. Li, J. Yuan, and D. W. K. Ng, “Performance analysis and window design for channel estimation of OTFS modulation,” *arXiv preprint*, 2021. [Online]. Available: <https://arxiv.org/abs/2101.11770>
- [13] M. F. Duarte and Y. C. Eldar, “Structured compressed sensing: From theory to applications,” *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4053–4085, jul 2011.
- [14] F. Gómez-Cuba and A. J. Goldsmith, “Compressed sensing channel estimation for OFDM with non-Gaussian multipath gains,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 1, pp. 47–61, 2020.